

IN THE SPECIFICATION:

Please amend the paragraph beginning on page 23, line 14 as follows:

Figure 9 illustrates another example of how the graphical representation of a piece of equipment may depend on the values of its variables. For example, a connector 210h connecting the outlet of a cooling load 213 to the inlet of a chiller 214 may be rendered in a color indicating that it is at a higher temperature than the fluid flowing through the connector 210j, which carries fluid from the outlet of the chiller 214 to the inlet of the cooling load 213. The color of connectors 210h and 210j may both be different from the colors used for connectors 210k and 210l used to connect a boiler 216 to a heating load ~~214~~ 218 (e.g. heating or cooling), providing a visual indication that connectors 210h and 210j are being used to connect cooling components and that connectors 210k and 210l are used to connect components of a heating system.

Please amend the paragraph beginning on page 31, line 11 as follows:

Referring to Figure 12, an analysis module 92 may include various executables, such as, for example, a validation module 286 and a calculation module 292. The validation module 286 may analyze the design data to determine unacceptable configurations or parameters. The connection checking module 290 may analyze the connections between components and provide feedback to the user indicating unacceptable connections. Unacceptable connections may include, for example, connecting the outlet of one component 124 to the outlet of another component 124. The connection checking module may function in conjunction with the error indication module 152 (Figure 4) to visually indicate errors on a computer screen or other output device 24. For example, in Figure 6, the breaks 199 in a connector 198b indicate that the outlet of a heating coil 184b is connected to the outlet of heating coil 194.

Please amend the paragraph beginning on page 34, line 5 as follows:

A CAD software interface module 310 may ~~enables~~ enable a user or a computer 11 to read directly the output data of a computer aided design (CAD) software package in order to acquire data concerning the interior spaces in a building designed with such a package (application). The CAD software interface module may automatically (or with user intervention) create components 124 or connections 126 based on the data output by the CAD software. For example, a building designed using a CAD software package may include descriptions of several rooms. The CAD software interface module 310 may read the description of the rooms, automatically create components 124 describing the rooms, and insert them into a project 116.

Please amend the paragraph beginning on page 37, line 13 as follows:

A project object may also contain components 124 and connections ~~336~~ 126, which may be embodied as instances of component objects 336 and connector objects 338, respectively.

Please amend the paragraph beginning on page 37, line 15 as follows:

The methods of a project object 332 may include, for example, attribute accessing methods 362, attribute editing methods 364, rendering methods 366, numbering methods 368, design updating methods 370, reporting methods 372, validating methods 374, design creating methods 376 and design editing methods 378. Attribute accessing methods 362 and attribute editing methods 364 may enable a user to access and edit, respectively, the attributes ~~370~~ 350 of a project object 332.

Please amend the paragraph beginning on page 37, line 21 as follows:

A rendering method 366 may function in conjunction with rendering methods 430, 450 of the component objects 336 and connector objects 338 it contains. Thus it may support display of a graphical description of a project object 332 on an output device 24. Numbering methods 368 may assign and store identifying data corresponding to the components ~~126~~ 124 and connections ~~124~~ 126 of a project object 332 when they are added to a project object 332. Numbering methods 368 may also enable a user to modify the identifying data of the components ~~126~~ 124 and connections ~~124~~ 126.

Please amend the paragraph beginning on page 38, line 6 as follows:

Design updating methods 370 may maintain the consistency of the ~~connections~~ components 124, connections 126, and design data 360, such that when some data is modified, other data that is dependent on it is updated to reflect the change. A reporting method 372 or methods 372 may gather information from the attributes 350 of a project object 332 to generate reports, such as cost summaries, parts lists, and the like. A validating method 374 may analyze the attributes 350 of a project object 332 and determine if there are any unacceptable design configurations or parameters. A design creating method ~~375~~ 376 may enable a user to insert components 124 and connections 126 into a project object 332. A design editing method 378 may permit a user or other objects to access and edit the attributes 350 of a project object 332.

Please amend the paragraph beginning on page 38, line 16 as follows:

Referring to Figure 17, the attributes of a shape object may have attributes 380 consisting of operational data necessary for the function of the methods 382. The methods 382 may comprise user interface handling method 388 and rendering methods 390. The methods 382 may be virtual functions which are defined by objects which inherit from the shape object 331. The user interface handling method 388 may receive and interpret mouse clicks, mouse movements, and the like. For ~~example,, a example,~~ a example, a user interface method 388 may move the graphical representation of a shape object to a different location on a computer screen based on the movement of a mouse by a user.

Please amend the paragraph beginning on page 39, line 3 as follows:

Referring to Figure 18, an equipment object 334 may inherit the methods and attributes of a shape object 331 as known in the object-oriented programming art. The attributes 334 of an equipment object 334 may, ~~For example,, comprise~~ for example, comprise notes 404, scheduling data 406, and product information 408.

Please amend the paragraph beginning on page 39, line 7 as follows:

Notes 404 may likewise, ~~For example,, comprise~~ for example, comprise any text a user may choose to associate with an instance of an equipment object 334. Notes 404 may also comprise data uniquely identifying an instance of an equipment object 334, notes may also be embodied as annotation objects 333 added by a user or associated automatically with an equipment object 334. Scheduling data 406 may comprise data needed for compiling reports or schedules about a project 116a, 116b. Scheduling data for an equipment object may include items such as cost, energy consumption, and the like.

Please amend the paragraph beginning on page 39, line 18 as follows:

The methods 402 of an equipment object 334 may include, ~~For example,, attribute~~ for example, attribute accessing methods 410, data calculating methods 412, data updating 416 methods and reporting methods 418. Attribute accessing methods may enable a user or even other objects to access the attributes of an instance of an equipment object 334. Data calculating methods 412 may calculate values for some of the attributes 400 of an instance of an equipment object 334 based on other attributes 400 of the instance.

Please amend the paragraph beginning on page 40, line 8 as follows:

Referring to Figure 19, a component object 336 may inherit attributes, methods, or both from an equipment object 334. The attributes 420 of a component object 336 may include, ~~For example,, rendering~~ for example, rendering data 424, connection data 428, type data 426 and user interface data 427. Rendering data 424 may include a graphical representation of an instance of a component object 336, its screen location, size and the like. Connection data 428 may include information indicating instances of connector objects 338 (Figure 20) connected to the component object 336. Connection data 428 may include information indicating other component objects 336 connected to a component object 336.

Please amend the paragraph beginning on page 40, line 16 as follows:

Type data 426 may include data indicating to which type of schematic an instance of a component object belongs. Type data 426 may also indicate what type of ~~equipment an~~ equipment a connection object pertains such as a pump, chiller, or the like. User interface data 427 may include information such as the screen location, size and the like of a component object 336.

Please amend the paragraph beginning on page 40, line 20 as follows:

The methods 422 of a component object 336 may include, ~~For example,, rendering for~~ example, rendering methods 430, updating methods 434, and attribute editing methods 432.

Rendering methods 430 may include methods that render a graphical representations of a component object 336 to a computer screen or other output device 24. Rendering methods 430 may also provide some of the same functionality as a mapping module 154.

Please amend the paragraph beginning on page 41, line 7 as follows:

Referring to Figure 20, the attributes 440 of a connector object 338 may include, ~~For example,, connection~~ for example, connection data 444, rendering data 446, flow data 448 and user interface data 449. Connection data 444 may include information identifying the component objects 336 that a connector object 338 connects. Rendering data 446 may also include data governing how a connector object 338 is displayed graphically. Rendering data 446 may include the screen location of a connector object 338, its shape, or the screen location of points it passes through. Flow data 448 may include information regarding the type or direction of fluid that is to pass through a connector object 338.

Please amend the paragraph beginning on page 41, line 15 as follows:

The methods 442 of a connector object 338 may include, ~~For example,, rendering for~~ example, rendering methods 450, attribute accessing methods 452 and attribute editing methods 454. Rendering methods 450 may include methods that render a graphical representations of a connector object 338 to a computer screen or other output device 24. Rendering methods 450 may also provide some of the same functionality as a mapping module 154. Attribute accessing and editing methods 452, 454 may provide for a user, or other objects, to access and modify, respectively, the attributes 440.

Please amend the paragraph beginning on page 42, line 1 as follows:

Referring to Figure 21, the attributes of ~~a information~~ an information updating object 330 may include, ~~For example,, type~~ for example, type data 464, and connection data 466. A data updating method 416 of an equipment object 334 may create an information updating object in order to determine values for attributes 400. A data updating method 416 of a project object 332 may analyze all the information updating objects 330 created by the objects contained therein and determine the information that each equipment object 334 needs.

Please amend the paragraph beginning on page 42, line 7 as follows:

The attributes 460 of an information updating object 330 may include, ~~For example,, data~~ for example, data necessary to enable a project object 332 to update the attributes of the objects contained therein. The attributes 460 may include, ~~For example,, type~~ for example, type data 464 indicating what type of data an object needs. For example, a boiler object may need to know the inlet temperature of the fluid entering it. Accordingly a boiler object may generate an information updating object with type data 464 indicating that it needs fluid temperature data.

Please amend the paragraph beginning on page 42, line 13 as follows:

Connection data 466 may indicate to which other objects an object is connected to. A project object 332 may use this information to determine the needed information. For example, a ~~boilers~~ boiler's inlet temperature is dependent on the other components 124 that connect to it.